CD-doc-2687-v1
May 17, 2008

CDF and D0 Tape Usage

Rob Kutschke, CD

Abstract

This document describes a very rough model of how CDF and D0 will use tapes in the next few years. It gives the background to the projections that are summarized on sheet "Tape Volume" of the spreadsheet. It is not a first principles model. Instead it averages over past usage, over a long enough interval to average out short term variations in usage patterns. It then extrapolates future usage by assuming that average use will not change a great deal and that the luminosity delivered by the Tevatron over January to April 2008 is typical of what the Tevatron will deliver between now and the end of the run.

Contents

1 Introduction

In the absense of a detailed bottoms up computing model, I have developed a model for future tape usage that extrapolates from past tape usage. The outline is:

1. Assume that January-April 2008 is representative of future normal running conditions. Predict raw data rates using the average over this period and account separately for the spring 2009 shutdown.

2. From somewhere I need the ratio of the volume of all data products derived from the raw data to the volume of the raw data; the derived data products include full reconstruction output, stripped reconstruction output, standard ntuples and physics group specific ntuples. For D0 this includes thumbnails and ntuples. I cannot get this from January-April 2008 since data production lags data taking by as much as a few months. Instead I get this by averaging the full 2006-2007 run; details below.

3. I have made an estimate of tape needed to hold MC data sets based on recent usage. The model is that this level will remain constant for both CDF and D0 in future years. In an earlier version of this note I had the required tape for MC data sets growing like the total data volume; that is no longer the case.

There is an important caveat. This model was created by looking at databases that describe the files that are now on tape. Any files that were written to tape and subsequently deleted will be missed by this model.

2 CDF

2.1 Method

Randy Herber developed an SQL query, listed in Appendix A, to extract tape usage statistics on a calendar month basis. His query produced a listing of the number of files written to tape, number of bytes written to tape, and the number of distinct volumes onto which these files were written; these listings were broken down by calendar month, and by 15 different categories of data. To simplify things I aggregated these 15 data categories into four major categories, as shown in Table 1

Figure 1 shows, for each calendar month since Jan, 2002, the amount of data of each major category. In this study, the time attached to a file is the time at which the file became known to Enstore. The upper left plot shows the time series of the volume of raw data. The upper right plot shows the time series of the volume of “reconstructed” data, where “”reconstructed“” includes all types of data products derived from the raw data. The lower left plot shows the same for all MC data types and the lower right shows the time series for generator level events. The vertical scales on the four plots are very different.

On the upper left plot the 2006 and 2007 shutdowns show clearly; vertical dashed read lines have been drawn to delimit the 2006-2007 run, from June-2006 to August-2007, inclusive. The sum of the raw data between these vertical lines is 315 TB. On the upper right plot, showing reconstructed data, the shutdowns are less apparent. As an approximation I guessed that the data from the 2006-2007 run was reconstructed between September-2006 and November-2007; that

Major Category	Original Category	Size (TB)
RAW	RAW	938.0
RECO	RECONSTRUCTED	1122.2
	NEW_RECONSTRUCTED	9.3
	STRIPPED_RECONSTRUCTED	49.0
	STANDARD_NTUPLE	164.5
	ROOT_NTUPLE	50.5
	HBOOK_NTUPLE	0.3
	TOP_NTUPLE	39.7
	B_NTUPLE	19.8
MC	SIMULATION	145.3
	SIMULATED_RECONSTRUCTED	342.2
	STRIPPED_SIMULATED	0.5
	STANDARD_NTUPLE_MC	95.0
	TOP_NTUPLE_MC	33.4
GEN	GENERATOR_LEVEL	3.7
Total		3013.4

Table 1: Grouping of data types for CDF. The left column gives the major categories used in this note; the middle column gives the categories, as defined in the Oracle database, that contribute to each major category; the right column gives the size, in TB of the data in each category, summed from the start of Run II. Here, as in the rest of this note, 1 TB=1024^4 bytes.

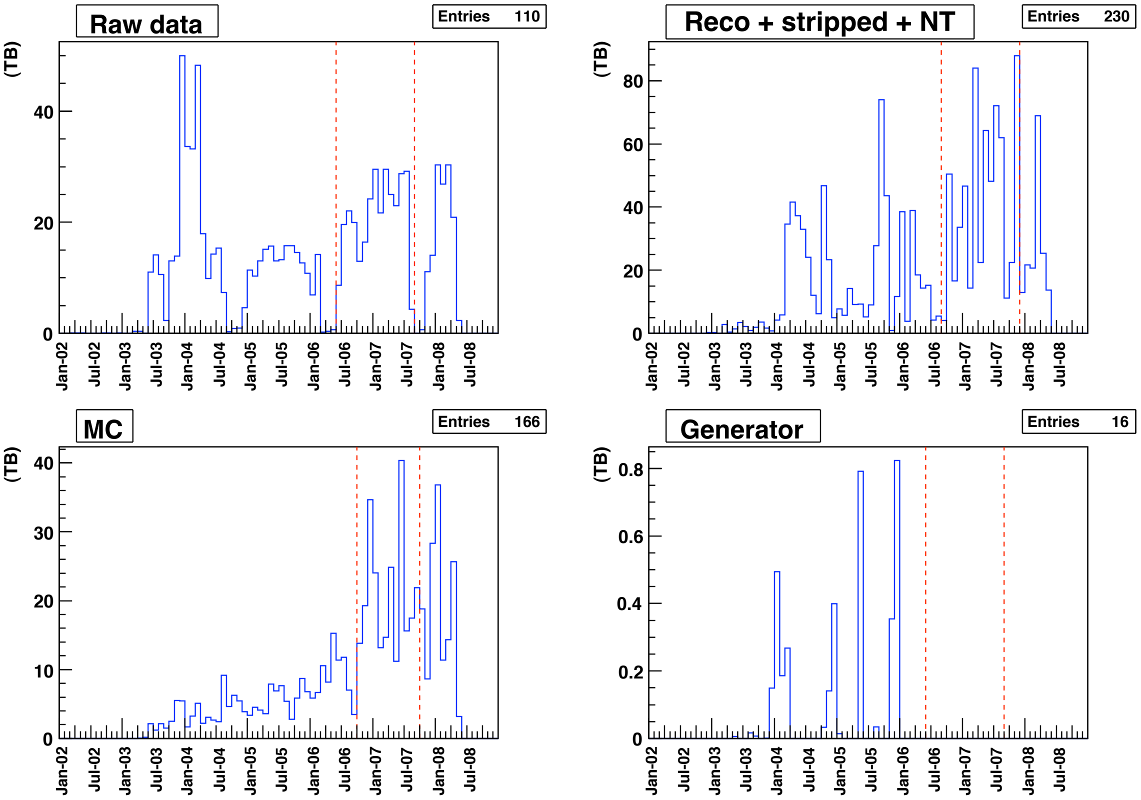

Figure 1: CDF Tape usage, by month, in TB. See the text for details.

is, the full reconstruction, including nutple making, follows the data taking by 3 months. The vertical dashed red lines on the upper right plot mark this shifted date range. In selecting this range, I am guessing the spike in the reconstructed data in November 2007 is the tail end of the 2006-2007 run, not the start of the 2007-2008 run. The sum of the reconstructed data written to tape in this time interval is 640 TB. The ratio of the total size of reconstructed data, to the size of the raw data from which it is made is 2.03.

Note that not all raw data is processed. Some of it is calibration or monitoring events and some of it is data of poor quality. So embedded in this model is the assumption that the fraction of unprocessed raw data does not change much from month to month.

The two vertical dashed lines on the lower left plot mark the normalization interval chosen for production of MC events. Somewhat arbitrarily, I chose this interval to be FY2007; it seems fairly representative. The volume of MC events written to tape in this interval is 251 TB. The old CDF model specfies that the number of MC events per year will be 0.7 times the CSL rate and that the size of an MC event will be 1.1 times the raw event size. For 2008 this predicts 260 TB of MC data from 335 TB of raw data. Since this is a crude approximation I projected the MC demand to be constant at 250 TB/year for the remaining years of the model.

The lower right plot shows the remaining category of events, the generated MC events. This shows that these data are no longer being written to tape.

2.2 Results

From above, the two key numbers are:

1. The ratio of volumes of all derived data to all raw data, 2.03.

2. The volume of MC data written to tape in FY2007, 251 TB.

These numbers are used in Table 2. The other input to the model is a contingency of 30%, which comes from 10% as a guess at the effects of higher occupancies due to higher luminosities and 20% general contingency.

The first row of that table is copied from the presentation made to Vicky on May 7, 2008, (on the sheet "Tev and DAQ"); it gives the volume of raw data projected to be taken in the given fiscal year. The next two rows of the table are computed in this model. The second row gives the projected volume of all derived data products for each fiscal year; it is computed as the number in the first row multiplied by the factor 2.03, given above. The third row is the projected volume of all MC data products. It is a constant 250 TB/year as discussed above. The fourth line gives the total of lines 2, 3 and 4; the fifth line adds 30% contingency and converts to PB (1 PB=1024 TB), which is the bottom line of the model.

	FY08	FY09	FY10	FY11
Raw Size(TB)	335.2	361.3	481.7	0.0
Reco Size(TB)	680.4	733.3	977.8	0.0
MC Size(TB)	250.0	250.0	250.0	250.0
Total(TB)	1265.6	1344.6	1709.5	250
+30% Contingency (PB)	1.61	1.71	2.17	0.32

Table 2: Projected Tape Requirements for CDF, by fiscal year. The table is discussed in the text.

3 D0

3.1 Method

For D0 a similar method was used. The main difference is that Randy Herber's Enstore usage tables were not available for D0 (they will be soon). Therefore the query was based on information in the SAM data tables; the query is given in Appendix B.

For D0, the role of the CDF data categories is played by two columns from the SAM data bases, the FILE_TYPE and the DATA_TIER columns of the DATA_FILES table. There are distinct 69 combinations of these fields with non-zero amounts of data but only 12 contribute signficantly to tape files created since July-2006. I have chosen this date as a cutoff since some of the FILE_TYPES and DATA_TEIRs were used earlier in D0 running but have not been used since. Table 3 shows the breakdown of the D0 data into 4 major categories, with meanings similar to those for CDF; the main difference is the last one, which I don't know if it is data or MC. In the following I will guess that it is data and will add it into the data sum.

Figure 2 shows the time series, by month, of the data volume written to tape in each of the 4 major D0 data categories. In the upper left plot, raw data, the vertical dashed red lines demark the 2006-2007 run. In the upper right plot, reconstructed data, the vertical dashed red lines demark the 2006-2007 run shifted to the right by 3 months (as was done for CDF). In this model, I presume that all reconstructed data created between the vertical dashed red lines corresponds to the raw data taken in the 2006-2007 run. This gives a ratio of reconstructed data volume to raw data volume of 1.58. Why is this smaller than for CDF? Part of it is because D0 does not write a copy of its raw data; but that only changes the question to, why is it bigger than CDF. I don't know the answer to that one.

The lower left plot of Figure 2 shows the monthly time series of the volume of MC data. As can be seen in Table 3, this is mostly imported thumbnail files and derived root-tree-by-group files. However the date rates are rapidly changing with time so these data are not enough to build a model; that will be discussed in the next subsection.

The lower right plot of Figure 2 shows the monthly time series of the volume

Category	FILE_TYPE	DATA_TIER	Size (TB) (After 07/01/2006)
RAW	importedDetector	raw	404.0
RECO	derivedDetector	filtered-raw	10.8
		raw-bygroup	1.4
		reconstructed-bygroup	2.3
		root-tree-bygroup	168.0
		thumbnail	456.7
MC	importedSimulated	simulated	8.1
		reconstructed	12.5
		thumbnail	70.9
	derivedSimulated	root-tree-bygroup	54.0
Other	physicsGeneric	reconstructed-bygroup	5.5
		.root-tree-bygroup	7.7
Total			1201.9

Table 3: Grouping of data types for D0. The left column gives the categories used in this note; the middle two columns gives the categories, as defined in the SAM database, that contribute to each category from column 1; the right column gives the size, in TB of the data in each category, summed from the start of July 2006. The reason for this choice of start date is that other categories of data were written at earlier times but not in recent times. D0 has a sixth FILE_TYPE, “nonPhysicsGeneric”; it contains only 0.4 TB over the interval in quesiton. Here, as in the rest of this note, 1 TB=1024^4 bytes.

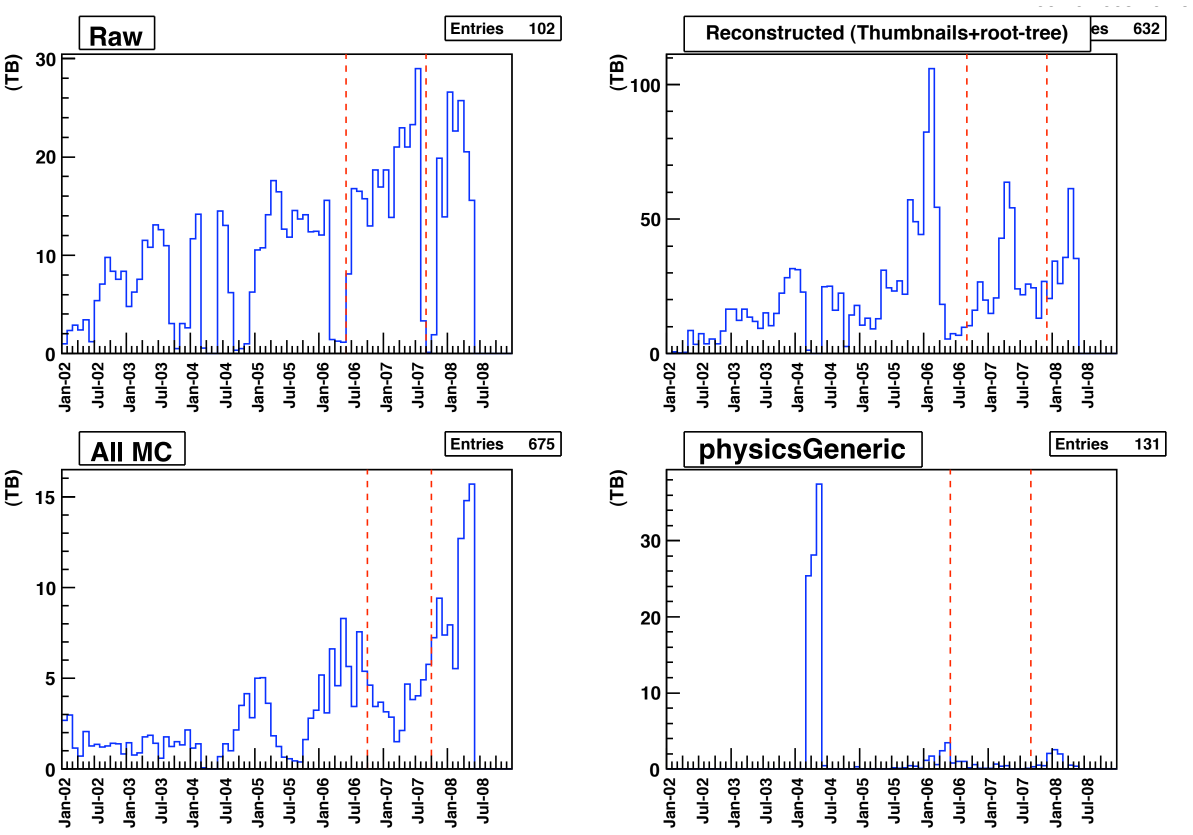

Figure 2: D0 Tape usage, by month, in TB. The vertical scales are different on all plots. See the text for details.

physicsGeneric data. Most of this data is outside of the scope of this discussion but a small amount of data has been written since 1-Jul-2006; that data has been added to the sum of all reconstructed data but it makes less than a 1% difference in the projections.

3.1.1 D0 MC Tape Requirements

Figure 3 shows the same information as Figure 2 but broken down into the six data categories that dominate the D0 tape usage. The MC related quantities are plotted in the second row. The lower left plot shows a time series of imported simulated data. This has not been an important contribution until recently. D0 expects to have a new reconstruction executable ready later this year. They are currently creating simulated events offsite and writing them to tape at FNAL; these events will be read back and reconstructed (offsite) once the new executable is ready. Based on an email discussion among D0 people in Feb 2008, it was projected that this would require about 200 LT02 volumes, or about 40 TB. This tape volume is a special requirement for 2008 only. The spike in events in the lower left plot is the leading edge of this effort.

The next two plots in the bottom row show the monthly contributions of thumbnails and root-trees-bygroup to the data volume. We need a few more plots to understand these data. Figure 4 shows the average event size, by month, for each of the important data categories. From the middle and right plots in the bottom row we can see that, for about one year now, the average size of an MC thumbnail has been about 90 kB per event and the average size of a root-tree event has been about 40 kB per event. I don't know if this will change with the new version of the reconstruction executable but I have included scale factors in the model to allow for such changes.

Figure 5 shows a time series of the number of events produced each month for each of the six most important data categories. From the lower middle plot one can see that D0 has been producing about 50 million events per month for the past 9 months. This corresponds to about 11.7 million events per week, slightly higher than the nominal D0 plan of 10 million events per week. The lower right plot shows a big burst of root-tree making in March and April 2008. In those two months, root-tree data was generated for about 450 million events; in the months before that, the root-tree making approximately kept up with the thumbnail making. I guess that in March and April a new an improved root-tree dataset was created from the exisiting thumbnails.

So here is my model of D0 MC production. D0 will produce 50 million events per month and, for each event, will write one thumbnail and 2 root-tree events, for a total size of 170 kB per event. This corresponds to 95 TB/year. I presume that this will continue indefinitely.

3.2 Results

From the preceding sections I need the following numbers:

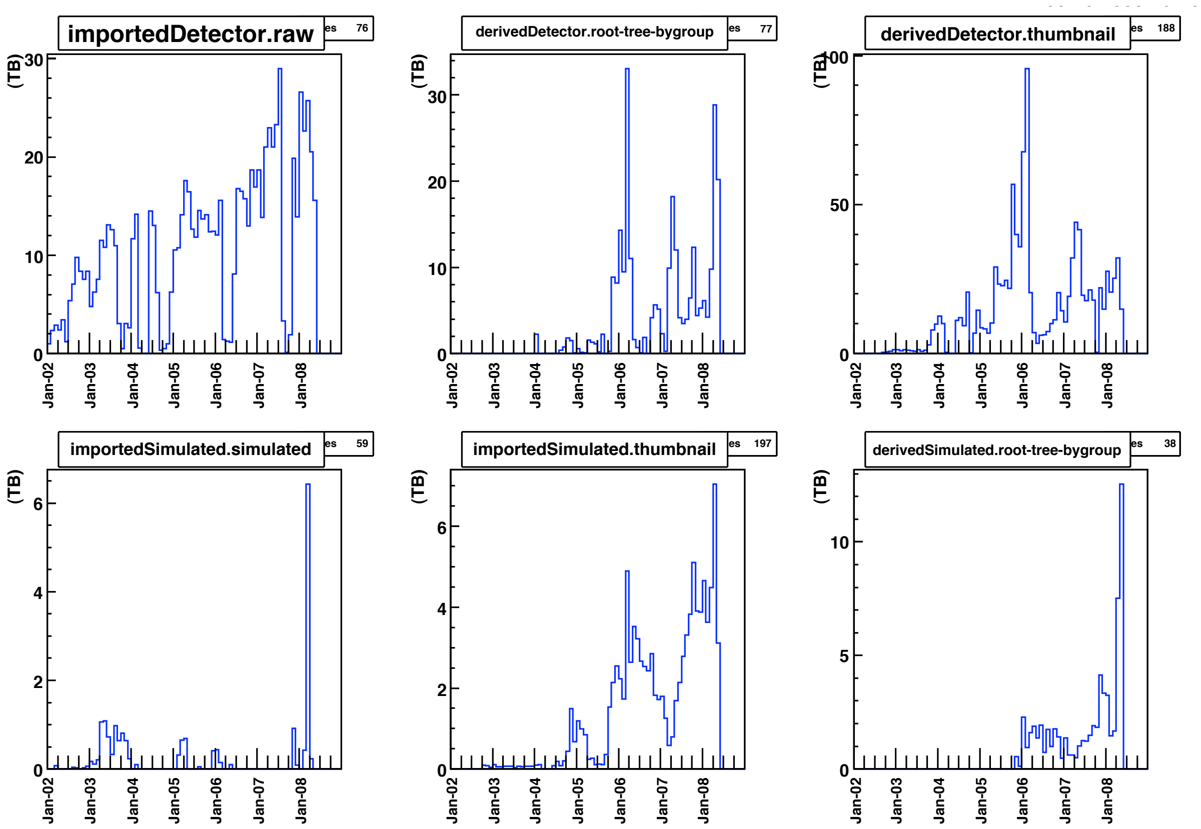

Figure 3: Detail of D0 Tape usage, by month, in TB. The vertical scales are different on all plots. See the text for details.

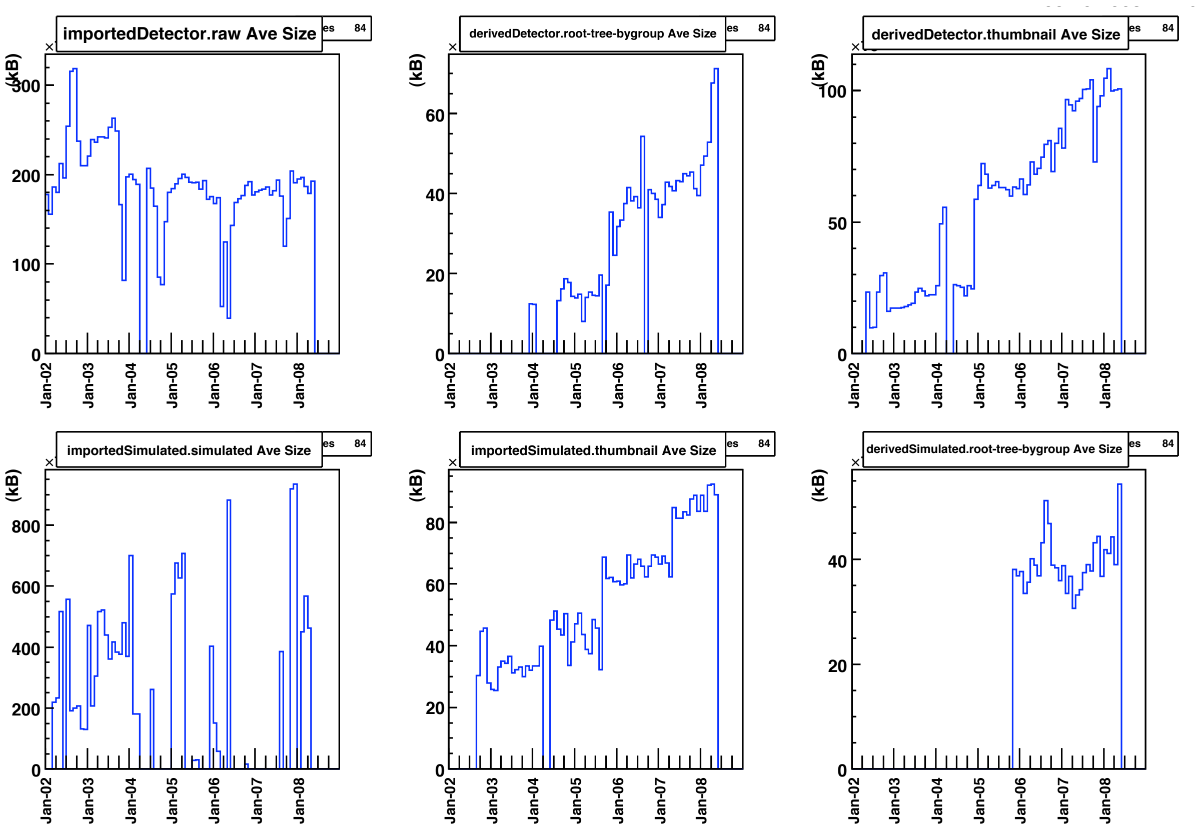

Figure 4: Average event size, per month, for the important data categories. The vertical scales are different on all plots. See the text for details.

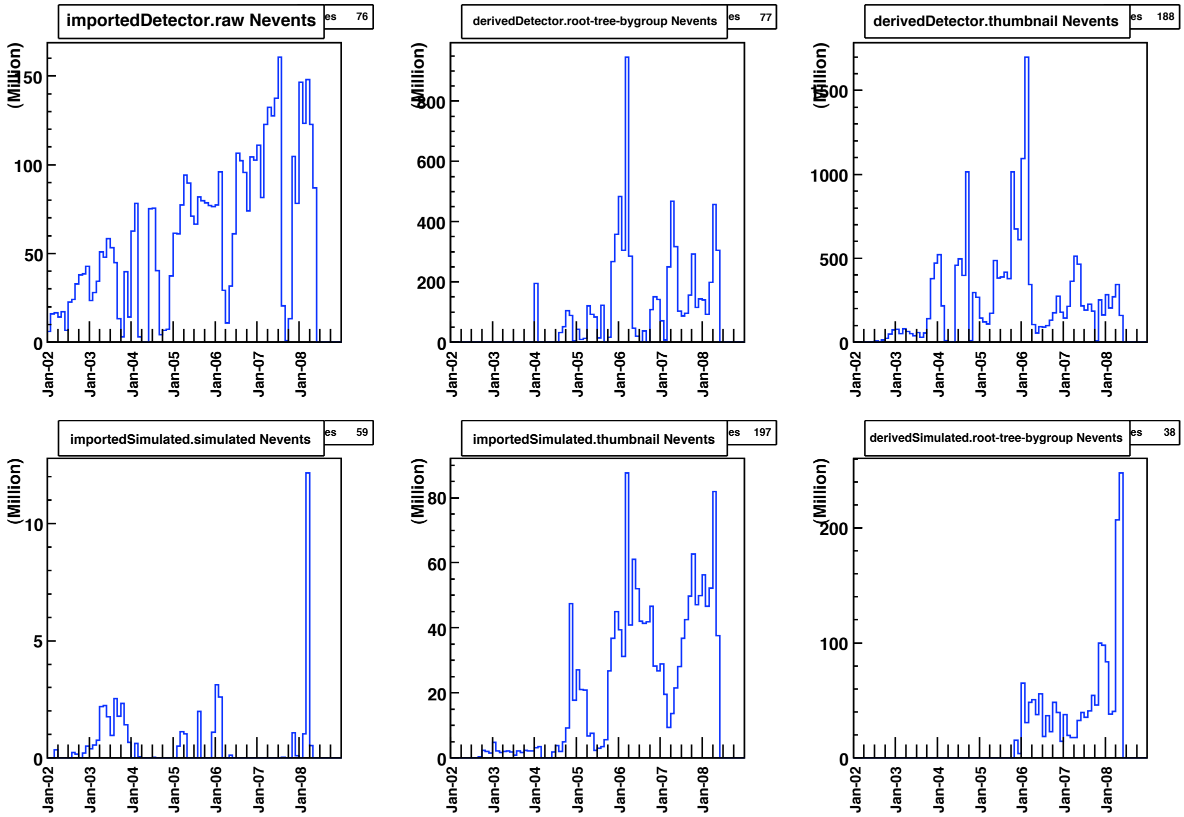

Figure 5: Number of events produced per month, for the important data categories. The vertical scales are different on all plots. See the text for details.

	FY08	FY09	FY10	FY11
Raw Size(TB)	284.2	301.3	401.8	0.0
Reco Size(TB)	449.0	476.1	634.8	0.0
MC Size(TB)	95.0	95.0	95.0	95.0
Buffer (TB)	40.0	0.0	0.0	0.0
Total(TB)	868.2	872.5	1131.6	95.0
+30% Contingency (PB)	1.10	1.11	1.44	0.12

Table 4: Projected Tape Requirements for D0, by fiscal year. The table is discussed in the text.

1. The ratio of volume of (thumbnail+root-tree) events to the volume of raw events, 1.58.
2. The volume of MC events written to tape per year, 95 TB.

The results are summarized in Table 4 which has the same format as the CDF table. The one difference is the row labeled “Buffer”, which is the space to hold the imported simulated events until the new reconstruction code is ready.

4 Summary

The summary of this report is given in Table 5. One should remember that this model is based on the following assumptions.

- Raw data rates and sizes, measured from Jan to April 2008 will be representative of rates and sizes for all data from now to the end of the run.
- That the reconstructed data from the 2006-2007 run is well represented by adding up all reconstructed data in a time window shifted 3 months later than the 2006-2007 run.
- That CDF and D0 will continue to produce MC events at about the rate and volume they are now.
- The model does not account for data that was written to tape and then later removed from tape.

	FY08	FY09	FY10	FY11
CDF (PB)	1.61	1.71	2.17	0.32
D0 (PB)	1.10	1.11	1.44	0.12

Table 5: Summary of projected tape requirements for CDF and D0, by fiscal year. The numbers include raw data, all data derived from the raw data, and all MC events. The numbers are in PB, where 1 PB=1024**5 bytes.

A CDF SQL Query

The listing below shows the sqlplus query that was used to generate, for CDF, the data used in this note. The sql file is included in this document as cdf.sql.

```
column ”#B” format 9999999999999999990
column ”#V” format 999999999999990
column ”#F” format 99999990
column data_type format a30
column library format a18
set linesize 132
set pagesize 49999
set trim on
set trimspool on
spool kutschke4.out
select sum(file_size) ”#B”,
       count(distinct volume) ”#V”,
       count(*) ”#F”,
       decode(grouping(to_char(pnfs_date,’YYYY-MM’)),1,’
           total’,to_char(pnfs_date,’YYYY-MM’)) made,
       decode(grouping(library),1,’total’,library) ”
           Library”,
       decode(grouping(fdt.description),1,’total’,fdt.
           description) data_type
from
  herber.en_volumes env,
  herber.en_files enf,
  herber.en_datasets eds,
  filecatalog.cdf2_dataset_registries fdr,
  filecatalog.cdf2_data_types fdt
where env.volume_id = enf.volume_id
  and enf.dataset_id = eds.dataset_id
  and eds.dataset = fdr.dataset_id
  and fdr.data_type_id = fdt.data_type_id
group by cube(to_char(pnfs_date,’YYYY-MM’),library,fdt.
    description)
order by made,library,data_type
```

```
/
spool close
```

B D0 SQL Query

The listing below shows the sqlplus query that was used to generate, for D0, the data used in this note. The sql file is included in this document as d0.sql.

```
column "#B" format 9999999999999990
column "Type" format a20
column "User" format a16
column "Tier" format a30
column "#F" format 99999990
column "#C" format 99999999990
set linesize 132
set pagesize 49999
set trim on
set trimspool on
spool bytype_ontape.out
select sum(file_size_in_bytes) "#B",
       count(*) "#F",
       count(EVENT_COUNT) "#C",
       decode(grouping(to_char(df.create_date, 'YYYY-MM'))
           ,1,'total',to_char(df.create_date, 'YYYY-MM'))
           made,
       decode(grouping(file_type_desc),1,'total',
           file_type_desc) "Type",
       decode(grouping(df.create_user),1,'total',df.
           create_user) "User",
       decode(grouping(data_tier),1,'total',data_tier) "
           Tier"
from
  data_files df
 join file_types ft using (file_type_id)
 join data_tiers dt using (data_tier_id)
 join data_file_locations dfl using (file_id)
where df.create_date > '01-jan-2002'
  and dfl.volume_id is not null
group by cube(to_char(df.create_date, 'YYYY-MM'),
   file_type_desc ,df.create_user ,data_tier)
order by made,file_type_desc ,df.create_user ,data_tier
/
spool close
quit
```